PRESIDENT'S REPORT

Elections are over. I was elected President for another year. I am going to really need your support this year.

Elsewhere in the newsletter is a report on the CES show by the president of the Michigan ACE group. I am in 100% agreement with his views. Some additional comments are: The 1200 has a better keyboard "feel" than the 800; the use of the special function keys to move the cursor around and "home the cursor" with use of the control key is a good idea and should speed up screen editing. The Word Processor is great. It was designed by Peggy Allen of Atari and written by Bill Robenson of Text Wizard fame. It goes beyond Text Wizard as mentioned in the CES article. You can embed escape codes in the text so even though some of the standard printing commands do not support non-Atari printers, you can still use ATARIWRITER on your particular printer. I played around with it and found it to be very user friendly. Also DataSoft's Spell Wizard can be used with it. We will have a demostration of ATARIWRITER at the Computer Expo.

Speaking of the SAN DIEGO COMPUTER EXPO, it will be held at the Convention Center on February 18-21. We will have a booth there to show off the great things that the Atari will do. We are hoping that someone from Atari will be down with some new software and possibly the 1200XL to show. Don Perkins still needs people to help with the booth. Give him a call.

I've just been informed that a new magazine will be coming out for the Atari. It is called HI-REZ. it will appear in April. The magazine is looking for articles and programming ideas for future issues. If you are interested in becoming   a contributor, give a call and I will give you more information.

Have any of you had problems with error 164? This error says that a file number mismatch has occured. Then you are unable to use that program anymore. Gary Sewell and Wes Newell of ACUGD have come up with a fix. "In DOS there is a routine that checks for file mismatches when loading a program. That routine can be easily be NO-OP'ed as follows. With the BASIC cartidge inserted, boot in the system. Then type POKE 4148,234: POKE 4149,234 (return). The 234 is a NO-OP instruction. Then go to DOS and write out the new DOS. This patch sould have no side effects with any good files, so use it on all your disks."

Eric Goez has volunteered to teach the Assembly language class. He will be getting in touch with those who signed up at the meetings. If you were not at the meetings and are interested, contact him at 449-6817. His Assembly class meets at 10131 Woodpark, Santee, the first Thursday of each month at 6:30 p.m. This is a 6502 Assembly class. Additional help will be given by Dave Story on interfacing with the Atari operating system.

Atari is looking for people to work in their computer camps this summer. You have to have at least a knowledge of BASIC or PILOT programming knowledge. I have a fact sheet if you are interested. You can call Patricia Tubbs at (408) 745-4899 for more information.

The 8th West Coast Computer Faire will be March 18-20 at the San Francisco Civic Auditorium and Brooks Hall. Admission is $15. If we can purchase 10 tickets, they will cost $11 each. Contact me if you are interested. We will also have a sign-up sheet at the next meeting. If you possibly can, do should go to this event—there is nothing like it for home computing.

EDITOR'S OUTPUT

Greetings from the keyboard. I think you'll find some interesting stuff inside, this month.

Let me extend my thanks to all our contributors. In a club of over 225 members, we have a core of active contributors who really do more than their fair share. You guys know who you are, keep up the tremendous output. A special thanks to club member Rick Reichert, who contributed an **unsolicited** hardware review on a memory upgrade for his 400. More, please! Questions, comments, reviews, news... MORE, MORE! (Pardon me, I have to wipe the froth from my mouth ... .)

This issue of the I/O Connector contains well over 30,000 keystrokes of information. Of special interest is Chapter One of Don Perkins' tutorial series on Atari in the Office. Read the CES review; Arlan is quite a wit! Ron Miller debuts his Forth Avenue this issue.

As Editor, I get to look over Atari newsletters from the entire nation. I thought you'd be interested to know that if there is what could be called a "national reaction" to the new 1200XL, it is a pair of shrugged shoulders and a "So What?"... . Meanwhile, Atari keeps patting itself on the back. With competition from the Commodore 64, I wonder what their sales will look like in a few months.

Storch appreciated all the comments (and fuss) but took a bit of offense at my calling him a cynic. He'll be around in a later issue, probably with some ideas on piracy and the American way.

CONTRIBUTE!

SEMINAR NOTICE

ATTENTION INVESTORS!

The Investors Special Interest Group of the San Diego Computer Society, and Microcomputer Special Interest Group of the San Diego Chapter of the American Association of Individual Investors, will conduct a seminar to inform investors and to demonstrate the advantages of using Microcomputers as aids in making investment decisions. Real Estate, Stocks, Bonds, and Commodities will be covered by successful investors and professionals. Computer programs will be demonstrated in addition to the lectures.

The two-day seminar is scheduled for the weekend of March 12 & 13, 1983. Fee is $75.00. ($65.00 if received prior to February 22).

For additional information call Don Van Ness at 487-6925.

INSIDE

NORTH COUNTY

Our January meeting was another full house with lots of computers at work. There were so many things happening that I didn't get to see everything myself. I did notice we have some real sharp kids (under 15 years old) who really get involved in these computers. Remember, if you come to our meetings, there is no agenda, no program. It is just people and computers. You can come and leave anytime between 6:00 p.m. and 10:30 p.m.

Our meetings are held on the third Tuesday of each month. So our next meetings are **15 Feb**, **15 Mar**, **19 Apr**, **17 May**, and **21 June**. Yes, in February and March our meetings are held before the San Diego meeting because the third Tuesday comes before the third Monday. Take note . . . Our meetings are held at **8081 Mira Mesa Blvd.**, which is a church across the street and just west of the Mira Mesa Cinemas.

COMPUTER WORKSHOP

Beginner's Class

This is a workshop for brand new (and not so brand new) Atari computer owners. We help familiarize you with your machine and teach you a little about the language BASIC. During the classroom part of our session, we will be talking about IF...THEN, ON...GOTO, and a method of getting IF...THEN...ELSE out of Shepardson (ATARI 8K ROM) BASIC. We will be meeting on Saturday, 83.02.11 (February 11th). Call Don at 479-6253 for place and time.

LIBRARIAN'S REPORT

Hard to believe that another month has slipped by, 1984 is just around the corner. Has anyone read George Orwell's novel of the same name? It will be interesting to see how many of his predictions have actually come true.

Due to some minor technical problems, the information/library line is down for the present time. For the duration, the phone will be answered by a human (either my wife, son, daughter, or myself). I will try to return all messages as fast as I can. Please don't call for an appointment after 9:00 p.m., as you will get a high-pitched tone from the club's BBS.

Some words about the BBS: All programs will be left on for two weeks before they are changed. This is to allow the maximum amount of people to use it. A note to first time users—on disk drive one (D1:) is a directory/explanation of the programs which are available on disk drive two (D2:). Drive one also makes available a modem program specifically suited to BBS use. New users should, upon logging on, copy the modem program to disk, and copy, to the printer, the **newuser, help,** and **local BBS numbers** files. Remember the BBS is open from 9:00 p.m. to 6:00 a.m. Monday through Friday, and may be used by calling the main library number between those hours.

After the computer show, this weekend, I fully expect that our club will expand by approximately 25-30 members. At this moment our membership stands at approximately 225; a fairly large group considering that our club has been alive for only 2½ years.

If you're interested, the club library has thirty-nine disks filled with all sorts of programs for members use. Please feel free to use this service. For those of you looking for a different type of joystick, with a rapid-fire feature, the February issue of Radio-Electronics (p.43) has a construction article for such a joystick. Check it out.

Some information on new programs:

BASIC ROUTINES for the Atari - a multifaceted program from Adventure International, with all sorts of routines that may be used in your own programming. The subroutines cover everything from joystick routines to creating text mode in graphics 8. At $24 it is worth twice that for the novice programmer.

ABUSE - Don't Ask Software has finally crammed Don Rickles into a disk. Play this yourself or better yet have your worst enemy play it. I promise that it will reduce them to a quivering bowl of Jello .

BAJA BUGGIES - Ever wanted to compete in the Las Vegas Mint 500 mile off-road race of the Baja 1000? This is the program for you. Grab your joystick, canteens, and get ready. You must finish ahead of the field with as few collisions as possible. A small guage shows where you are in relation to everyone else. From Datasoft for $39.00

DRAWPIX from Artworx - a program which, with a little effort, will transform your screen into a work of art. The program works with Graphics 8 pixils which are unbelievably small. It is fairly easy to use, if you follow the instructions provided. $24.95

That's all for now, see you at the Computer Show.

Paul Caesar's
BEST BUYS

RETAIL SOURCES IN THE
SAN DIEGO COUNTY AREA

This month is a reprint of the revised source list for San Diego County Atari products and discounts. I remind you that many different kinds of stores are represented here and the quality of service varies. You can use this list as a guide to finding products you want and you may occasionally information at these stores. One of the most helpful is On-Line Computer Center.

A short time bargain is available from the On-Line people. They have the Atari 800 with 48K RAM priced at $400 while their stock lasts. I wish I had waited to buy mine from them, but even a bargain shopper sometimes misses. Another store which seems ready to help is MailComp; their store just opened.

I haven't heard from anyone out there - is this information helpful to you, am I missing any noteworthy stores, **does anybody read this?** If you have hints, suggestions, complaints about a dealership, or want the latest Retail Source list, send me a note. **Paul Caesar, 377 Fashion Valley, San Diego, CA 92107.**

HARDWARE REVIEW

Dynamic Technologies 48K Memory Upgrade Kit for Atari 400

With its recent price decrease to $109.⁹⁵ (plus $2.⁰⁰ shipping), I finally bought the Dynamic Technologies 48K memory upgrade kit for the Atari 400. The full-page ad in December '82 **Compute!** was the final nudge. I picked up the phone and called in to order. One week later a small, heavily padded envelope appeared in my mailbox. What a nice kit! No parts missing! Excellent instructions with plenty of drawings to show what goes where. Best of all, it worked first time up!

Screen clarity is very good, with only the faintest interference lines present. My wife and children noticed no change at all from the "stock" 16K board. A simple two-part memory test program in Basic checked the first 40K. No problems. As the Basic cartridge uses the last 8K, it could not be verified.

The kit consists of eight 64K RAM chips, solder, wires and instructions. Also included is a small circuit board with two chips on one side and peel-off sticky tape on the other. Assembly involves cutting certain etches on the 16K board, sticking the Dynamic Technologies circuit board onto the 16K RAM board, trimming, stripping and soldering about a dozen wires and replacing the 16K with the 64K RAM chips.

A Few Tips to Others Who Install This Kit:---

■ Don't stick the add-on board to the RAM card before soldering the first lead, even though the instructions tell you to do so. The add-on card blocks access to the solder point, making it difficult to get wire, solder and iron tip in there.

■ Buy (or borrow) a heat stripper. The insulation on the supplied wire is tough!

■ Etches must be cut on both sides of the RAM board. Make sure you're working on the **right** side at the **right** time!

■ Take care to get rid of any static electricity build-up before handling the RAM chips. You may want to perform "RAM transplant" on the kitchen counter, where you can reach over and touch the faucet to be sure you're electrically neutral.

■ Some of the soldering involves IC pins on the foil side of the RAM and main computer boards. Be extra careful to avoid excessive heating at these points. If you're not sure how much is too much, obtain knowledgeable assistance.

This kit was great fun and a modest challenge to install. It took me 2½ hours, which included one dumb mistake. (I cut one etch line which I shouldn't have—easily corrected).

If saving money is one of your objectives, I highly recommend the Dynamic Technologies kit.

SECRETARY'S REPORT

January Meeting

Jim Keese from On-Line announced their ACE discount policy: 10% off for purchases under $100, 20% off for $100 and over. Due to an over-stocking problem, they are also selling the Atari 800 with BASIC for $449 to members only.

Elections:

For Treasurer: Jim Marr was nominated from the floor, but declined. Tony Tait had been previously nominated by the nominating committee, and Dave Story was also nominated from the floor. Tony Tait won the office by default.

For Secretary: Don Perkins was chosen by the nominating committee and was put into office again for his third year.

For Vice President: Ted Langlett was chosen by the nominating committee. Dick Hiatt and Dave Story were nominated from the floor. The vote went to Ted.

For President: Dick Hiatt was nominated by the committee and also began his third reign by default.

The club voted to become a SIG of the San Diego Computer Society, and allocated $15 to pay for Dick's membership in the Society to qualify us. Warren McKenna, from the Society, told us about the new hard disk based BBS being put into the Reuben H. Fleet Science museum. It will be possible to put our programs from our library on-line there to access through a modem.

$100 was allocated for any necessities that the upcoming computer show would require. Another $25 was given to Chuck for the BBS.

Our scheduled speaker from Freedom Won, an Atari distributor with a different sales technique, cancelled out on us. We did hear from Rich Tier and friends from the new Orange Micro store on Clairemont Mesa Blvd., in Kearny Mesa. They spoke to the group about the history of Orange Micro, a printer store chain, and introduced the Gemini and Smith Corona printers.

There was some controversy over the article "Storch" which ran in the January "I/O Connector", centering on club representation vs. freedom of opinion. There will probably be more to come on this later...

The meeting ended with demos of Leading Edge's parallel disk drive system; Earthquake (another production by one of our own members—Machine language with fine scrolling graphics—looks great!); a FORTH language system—in ROM; and the game Baja Buggies.

NAME	ADDRESS	PHONE	HARD-SOFT	DISCOUNT	COMMENT
A.S.C. Moblie TV	7436 University Ave., La Mesa	284-5615	Both	15-20%	Repair Station for Atari
ABC Products	8868 Clairemont Mesa Blvd., S.D.	268-3537	Supplies	No	Low Prices - Disks, Cassettes, Etc.
Apple Country	P.O. Box 1099, Julian, CA 92036	765-0239	Both	25-30%	Mail or Phone order
Ardan	880 Arnele Ave., El Cajon	579-7722	Both	10-15%	Atari 400 & Software
ATV	8464 Commerce Ave., San Diego	578-5516	Repair	No	Repair of 400, 410, 800, & 810
B. Dalton Books	All Locations	See Book	Books	No	Software at Mission Valley
Best Products	7938 El Cajon Blvd., La Mesa	698-7244	Both	10-15%	Atari 400 & Software
Border Software	1196 Third Ave., Chula Vista	427-5085	Soft	25%	Phone or Mail order
Computer Age	4688 Convoy Dr., San Diego	565-4042	Both	10-15%	Atari 800 & 400
Computer Store Int'l.	1249 Broadway, El Cajon	579-8066	Both	10% W/Card	Discount on Non-Sale Items
Computer Stores Int'l.	329 Mission Ctr. West, San Diego	291-9531	Both	10% W/Card	Discount on Non-Sale Items
Consumer Computer	8314 Parkway Dr., La Mesa	465-8888	Both	10% W/Card	Discount on Atari Related
Dimensional Software	3954 Clairemont Mesa Blvd., S.D.	275-4243	Both	None	Large Selection of Software
Disney Electronics	4341 Twain Ave., San Diego	281-0285	Both	5-30% Card	Adding more Software
Discount Depot	Price Bazaar, 1440 Broadway, C.V.	691-1284	Both	15-20%	Atari 800 & 400
Fedco	1725 N. Euclid, San Diego	262-2411	Both	"Good"	"Large Selection"
Federated Group	3146 Sports Arena Blvd., S.D.	223-5301	Both	Will Deal	Show Low Price - Ask to Better
Kirk Paper	9737 Aero Dr., San Diego	268-3282	Supplies	No	Low Prices - Paper, Labels, etc.
MailComp	9434 Chesapeake, San Diego	277-8002	Both	20-25%	Very Helpful
On-Line Computer Ctr.	I-8 & Jackson, La Mesa	463-0397	Both	10-20%	Repair Station - Nice People
Pacific Stereo	All Locations	See Book	Both	10-15%	Watch Paper for Sales
Sierra National Corp.	5037 Ruffner St., San Diego	277-4810	Both	10-15%	Atari Repair Station
Softwaire Centre Int'l.	4170 Convoy St., San Diego	576-1424	Soft	None	Good Selection - Book, Mags
Super City	All Locations	See Book	Supplies	----------	Low Prices - Disks, Paper
The Broadway	All Locations	See Book	Both	No	Atari Products Only
The Computer Store	209 3rd Ave., Chula Vista	422-6785	Both	5-30% Card	Also See Disney Electronics
Toys-R-Us	All Locations	See Book	Both	15-20% Off	Atari 400 and Software Only
Video 2000	1455 University Ave., San Diego	299-1251	Both	Unknown	----------
Video Concepts	260 Fashion Valley	296-8330	Both	No	Atari 800 & 400
NAME	ADDRESS	PHONE	HARD-SOFT	DISCOUNT	COMMENT

SPECIAL REPORT

Atari in the Office: Introduction
#1 in an endless series

By Don Perkins

Although the Atari Computer is designed with the home user in mind, it is slowly creeping into the business world. I myself am employed as an Atari programmer for a small business in Pacific Beach. With the design of the Atari, it can fit into many surroundings. It is the business aspects of Atari design and programming that this series will focus on.

Throughout the series, I will use Atari (Shepardsen) BASIC for most programming examples, as it is as much an Atari standard as anything else. If you are not familiar with BASIC, these articles may be somewhat advanced for you at times. Conversely, if you are an "old pro", you will often find the subjects discussed very elementary. But if everyone will bear with me, I will try to be as explicit as possible, in order that all might benefit.

The most basic system that is required to do any business functions is the following:

    Atari 800 computer with BASIC
    Monitor capable of displaying 40 columns
    Atari 810 or other compatable disk drive.
    Printer and necessary interface.

A business micro-computer system cannot work efficiently when it is cassette based. A disk drive is required because it is able to handle large amounts of information quickly and without requiring inaccurate and time-consuming user intervention. A printer is an absolute must!

I will assume your system has these basic units if you are serious about any business programming.

The Disk Drive

The disk drive is an integral part of any business system. It loads the programs, updates records, stores customer/employee histories, stores payroll records... in general, it is the place where all the storage work is kept and retrieved. Disk drives are under complete control of the computer. Unlike a tape, where a human must fast-forward or reverse the tape by hand to the desired information, the disk drive can go to any section of the disk that the computer desires to be at. This makes it much more accurate and efficient than a tape.

The disk drive is usually used as a sort of filing cabinet, where information is stored in files. Each file has a name. The computer forces certain restrictions on how these files are named. Each file name is requred to start with a capital letter. The rest of the letters may be either capital letters or numbers, but nothing else. The name can be no longer than eight letters. Thus, MYFILE and FILE1 are "legal" (allowable) names, but 1ACCOUNT, FILE#1 and JIMJOHNSTONSFILE are not. The file name is also allowed to have an EXTENDER. That is, you may put a period after the file name, followed by one to three capital letters or numbers. FILE.1 or SMITH.DAT are examples of file names used with extenders.

Since the disk is so important, I will introduce a few of the commands for using the disk drive from BASIC.

The first command is OPEN. Before you can use a file in any way, you must first OPEN it. (Just like a file cabinet: you must first open the drawer before you can put something into or take something out of a file.) The OPEN command must be expressed as follows:

    OPEN # file, a, b, "D:FILENAME"

Where:
file	is a number from 1 to 7	
a	is a special code number	
b	is another code number, but will normally be 0 for our purposes	
filename	is a file name as described above	

The code numbers we will use are 4 and 8. 4 tells the disk you want to READ information. 8 says you want to WRITE information. Thus, OPEN #1, 8, 0, "D:TEST" opens file #1 for writing (output) and gives it the name TEST. Go ahead and type this in.

The next command is PRINT. As you may know, PRINT (or ?) puts information on the screen. If we say PRINT # file, we can put information into that file. Try this:

    PRINT #1, 500

You have just put the number 500 into file TEST. Notice that the disk drive didn't do anything yet. Don't worry. I'll explain why not in a later article. For now, trust me, the information will get on the disk.

Another command is CLOSE. We close a file when we are done using it. Type: CLOSE #1

You will notice that the drive came alive for a few seconds. Now type this:

    OPEN #1, 4, 0, "D:TEST"

Remember, this opens file number 1 for reading (input) on file TEST. Now type:

    INPUT #1, THIS

As you should know, INPUT accepts information from the keyboard. INPUT # file will get information from the file. Now do this:

    PRINT THIS
    and you'll see:
    PRINT THIS

    500
    READY

Do you get the idea? Experiment with these commands a little. Next month I'll be talking to the advanced users about data compression and give the new ones a program for keeping track of addresses. Until then, Happy business computing!!!

Frankly, I've never had any problem with Hexadecimal!

LAS VEGAS C.E.S.

The Winter Consumer Electronics Show (CES) was held in Las Vegas, Nevada, January 5th through the 9th. The show was truly massive this year. Even though the Vegas show is supposed to be the little sister of the summer Chicago show, this one felt bigger than the windy city's last CES.

The Big News (yawn)

As expected, the 1200XL turned out to be a very fashionable styling job (I checked underneath for the designer label...). Dressed in elegant white, black and silver, the machine is sure to be described by some loonies on the coast as the hottest item since electric shoelaces.

The **I/O Connector** scooped most of the 1200's major features last issue. Here's what was missed: No BASIC cartridge is included at the announced list price of $899. Atari has taken pains to insure upward compatibility. 400/800 software that conforms to the specs laid out in the 400/800 technical users notes should have no problems running on the new system. On the other hand, the unit is completely sealed, so leave your Ramdisks and Bit 3 80 column boards attached to your 800. The keyboard does have a very different feel, and your right pinky doesn't have to dodge the Atari key when you shift to upper case while touch-typing.

The START, SELECT, OPTION, BREAK, SYSTEM RESET, and programmable function keys, now moved above the QUERTY keys are flat chrome, touchsensitive squares. One of these keys is the HELP key, which can be used to access the internal diagnostics. I saw the memory, operating system, and keyboard diagnostics run. No errors occured in the units on display. I wonder what the error messages say—something like "RAM FAILURE at location 52E0 - TAKE ME TO A SERVICE CENTER - BRING YOUR CHECKBOOK"?

Checking the rear of the machine, the DIN plug is still there for driving composite color monitors and B&W sets, but sorry, it's still a 40 column display.

So what makes this machine better than best? According to Marketing, the GTIA has been reworked, offering more saturated and pure colors. The POKEY has been overhauled, with cleaner sound as a result. The CPU, although still a 6502, is a custom VLSI job that cuts down the number of discreet components and chips, which should, I am told, mean higher reliability (wanna bet...). It's a nice package, but at $899? $599 is a more realistic and competitive figure. And I'm still miffed about no access to the internal Bus.

The 800 Price Cut

The 800 list price was dropped to $679 for a 48K unit, a price which does not include the BASIC cartridge and user manuals. Somewhat disturbing is the rumor that newly manufactured 800s will have all 48K on a single board and have no connectors in the last two RAM slots and right-hand cartridge slots, to keep costs down. Hopefully these rumors are false. Hang on to your old 800.

In addition to the 1200XL, Atari unveiled the Model 1010 Cassette unit. It is nicely styled in the manner of the 1200. An Atari representative, questioned about the difference between the old 410 recorder and the new 1010, said "This one works!" The 1010 will list for $95.95. Enhancements of merit include I/O connectors designed around the Molex jacks, just like the 810 disk drive. This means that the 1010 does NOT have to terminate the serial daisy-chain. The 1010 also has an on/off switch and an LED pilot light. The 410 Recorder will be allowed to quietly disappear after the introduction of the 1010.

New Software

Atari unveiled their new **ATARIWRITER** word processing package at the show. It is one of the finest software products for the 400/800/1200 machines I have seen. The $79.95 16K cartridge appears to be extremely well written and thought out. Featurers include: Global search and replace, ability to format to the screen (while the normal display does not scroll, if you opt to format your text to the screen, horizontal scrolling is enabled. The formatted material may be up to 132 characters wide), Font selection, User Definable Tabs, Standard Atari DOS files, Works with Cassette, Right Justification, Auto Paragraph Indent, File Chaining, Double Columns, Auto Page Numbering, Headers & Footers, Centering, Form Letter Features (no, it's not a true file merge—it will halt at preselected points in your text while printing and allow pertinent data to be entered). I must admit I was truly impressed with this package. Atari refused to deny that the cartridge was done by the author of Text Wizard. It certainly had the ergonomic "feel" of the popular WP package, and DataSoft was biting their lip when asked if Text Wizard 2.0 would ever see the light of day. I feel the 2.0 version will never be released. No matter, **ATARIWRITER** will surely be one of my favorites. Now the bad news..., it supports only Atari printers out of the box. Some members of the Marketing staff just refuse to "officially" acknowledge the existence of the real world and they kiboshed plans for support of the NEC, Epson, ProWriter and other popular printers being included. Well, fear not! Some level heads did prevail, and support for those printers will come in the form of a disk of printer drivers for those devices that will be distributed through APX (the Atari Program Exchange). It will cost 20 or 30 dollars which is a tad ridiculous, but at least the support will be available.

ATARIWRITER is my choice for Atari CES honors based upon what I have seen so far.

In addition to the new WP program, a slew of new Atari Inc. software was shown.

On the educational scene for the very young, we saw **Juggles' House**, a program that teaches the concepts of upper/lower and inside/outside, and **Juggles' Rainbow**, which handles above/below and left/right. $29.95 each on disk or tape. Sesame Street seems as effective, a lot cheaper, and more entertaining to this kid.

For the 7 to 10 year-old crowd, Atari displayed **Mickey in the Great Outdoors**. Featuring well done graphics, the Ubiquitous Mouse of Disney fame teaches language arts and basic arithmetic. I liked this one, but the $50 price tag may give some pause. Available on disk or cassette.

Four new entries in Atari's personal development line popped up at the show. **Atari Speed Reading** comes on five cassettes at $79.95 the set. Evelyn Wood must be quaking in her boots. Some smidgens from an **Atari Music Tutorial** program were shown. No price was announced.

Home management buffs can look forward to **Timewise** ($30), a time management program on diskette that will help computer fanatics schedule the three minutes a day they spend speaking to their spouse.

The technically well-executed but tragically flawed **Telelink II** cartridge was demonstrated. For all its bells and whistles, Telelink II continues in the incredibly oblivious-to-user-needs Telelink tradition of not being able to dump data to disk or tape. This one will make an excellent doorstop or hockey puck for serious telecommunications buffs.

Ahhh..., what would life be without new Atari game software?

Although it had some cute graphics, the **E.T. Phone Home** game seemed a crashing bore to me. I thought that after divestiture of AT&T, direct dial was supposed to get cheaper. $50.

As if one movie tie-in wasn't enough, you can also look forward to **Superman III** ($50). At that price, Valerie Perrine isn't even included.

I'm not sure if **Star Trux** (rhymes with Bucks) is any better than the previous two. I'm sure the title resulted from Paramount pictures refusing to sell Warner Communications the rights to use the Star Trek name. Might serve the Legal department right if THEY got slapped with a subpeona for a change. If I paid $45 for the cartridge, I'd probably want to sue also.

Just so I don't get accused of being a negative thinker, there is a rose amoung the stinkers. **QIX** is a **very** nice translation of the arcade game. The graphics are nice, the play values are right, it has the feel of a winner. $45, but worth it in this case.

The Las Vegas CES officially broke all previous attendance records for **ANY** kind of trade show; over 80,000 people were there on Saturday!

BURST IO

by Pat Warnshuis, Editor, Portland Atari Club Newsletter,
3116 S.W. Evelyn St., Portland, OR 97219

(This article on Burst IO has been reprinted from the Portland Atari Club Newsletter, June, 1982 issue.)

BURST IO (input-output) is an extremely fast method of transferring block data from cassette or disk to the computer memory or from the computer memory to your storage device. Burst IO (aka block IO) is not supported by ATARI or Microsoft BASIC. BASIC supports either single string transfers or the GET and PUT bytes of data. However, to save a graphics 8 screen to disk using PUT in a FOR. . .NEXT loop requires some three minutes. Using Burst IO you can dump a screen to disk in less than three seconds.

Where will you want to use Burst IO? Anywhere you have a large block of data to move in or out of memory. For example, you can save and retrieve any screen in any graphics mode. You can save, load or edit complete character sets. You can predefine player-missile data, then save and load them using Burst IO. You can load contiguous blocks of data or complete data base management files. If you saw the introductory program for the PAC program library disks with the PORTLAND ATARI CLUB banners flashing across the screen, then you saw an application of Burst IO. In that routine the one thousand bytes of player-missile data was loaded from the disk so fast nobody even noticed the disk was turned on!

First, a little background. (I'm going to talk about writing to a disk. The procedures are the same for writing to a cassette.) When BASIC writes to the disk, it only writes in segments of 128 bytes. It never writes a single byte as is implied by the PUT command. When you issue a PUT command, BASIC takes that single byte and adds it to an IO buffer of 128 bytes. The buffer is merely a part of your memory which is dedicated as a temporary accumulator where BASIC collects your individual bytes for an eventual sector write operation. (A sector is a block of 128 bytes on the disk or a record of 128 bytes on the cassette.)

BASIC keeps adding individual bytes to the buffer until the buffer is filled. When the buffer is filled, it writes the contents of the entire buffer to the disk in one operation. The only way you can write a partially filled buffer to the disk is to CLOSE a file or END a program. This is why you must CLOSE a file or END a program to get the last segment of your data saved.

Similarly, when you do a 'GET #3,A' BASIC does not read a single byte from the disk. Rather, on the first GET command, it reads an entire sector of 128 bytes into the IO buffer. Then it sets a pointer to the first byte in the buffer and is ready to move single bytes from the IO buffer to where you want them. When the pointer goes beyond 128 bytes, BASIC knows it must read in the next complete sector and reset its pointer to the start of the IO buffer before it can do the next GET instruction.

In BURST IO, however, BASIC and the operating system do not use the IO buffer. Instead, you simply tell the Central Input-Output (CIO) routine in the operating system where the data starts in memory and how many bytes to write. BANG! The operating system moves the data out in 125-byte chunks directly from memory to the disk. (I'll explain why it's 125 byte chunks instead of 128 byte chunks later.) Reading is even simpler in burst IO. You tell the CIO to read 65,000 bytes of data and tell it where to start storing the bytes in memory. When it gets to the end of the file, it stops reading.

Let's say it again, for emphasis: You simply tell the CIO where to start saving or loading the data and how many bytes to transfer.

The operating system does the transfer and returns the status of the operation, either A-OK or an error number, in a memory location which can be PEEK'd by your program.

How do you do it? Well, first OPEN a file for read or write. BASIC then sets up a block of control data for the file. You can have eight files open at a time. BASIC keeps track of what you are calling each file, where the buffer for each file is (or where the start location is for Burst IO), whether it is a read or write operation, the number of bytes involved, and some hardware-peculiar information which you tell it about using the AUX1, AUX2 bytes in your OPEN statement.

Oh, yes! It also keeps track of where the machine language routine is for getting in and out of the particular hardware which you are accessing: cassette, disk, printer, modem, CRT screen, etc. That's why you can talk to all your hardware devices the same way. CIO only keeps track of WHERE the hardware driver is, not how it talks to the specific device. When the boys in the back room at ATARI wrote the rest of the operating system, they wrote the machine language routines for getting in and out of the standard devices, such as the disk, cassette, printer, etc. Then they made a table, called a Device Handler Table, which tells the CIO where each handler is. You can add your own handler routines and add them to the table. For example, if you have an odd-ball printer, or teletype machine, or a digitizer pad, or another computer, or any device at all, you patch in your code for talking to the device and add an entry in the device handler table to tell the CIO where the code is. Then you talk to the device the same way you talk to a disk, the CRT, or a printer. That is, you simply use GET/PUT, PRINT #/INPUT #, or burst IO.

BASIC needs 16 bytes of data to keep track of each file which it has opened. For our purposes, FILE 0 data starts at location 834. Call the location IOCB (Input/Output Control Block) for File 0. Then IOCB #1 is at IOCB + 16, IOCB #2 is at IOCB + 32, etc. At our IOCB# + 0 we POKE the direction command for our transfer: 7 for read, 11 for write, 12 for append. At IOCB# + 2,3 we POKE the start address for the data. IOCB# + 6,7 gets how many bytes we want to transfer. That's all!

Then we do a little machine-language call routine to jump directly into the operating system's CIO routine. This is the string you see so often which looks like this: BURSTIOS = "hhh*LVd". The '*' and 'd' are in inverse video. Once the IOCB# data is set up, we initiate a burst IO by the statement: "XFER = USR(ADR(BURSTIOS),16*CHAN#". where CHAN# is the file number we used in our OPEN statement for the file. The BURTSIO$ routine moves the CHAN# to the hardware index register and jumps to the operating system's CIO routine which does the burst IO.

Let's take another look at the screen editor which we ran in the June newsletter as an example of saving or loading text screens.

```
1 DIM F$(15):OPEN #3,4,0,"K:":OPEN #2,12,0,"E:":POKE 709,8:POKE 710,194
2 ? "Read or Write ":GET #3,C:IF C = 82 THEN A = 4:W = 0:GOTO 4
3 ? "NEW or Add ":GET #3,C:A = 8 + (C = 65):W = 1
4 ? "FILE ";:INPUT F$:OPEN #1,A,0,F$:IF A = 4 THEN 8
5 ? "} ± C Creates next screen":? " ± E Exits now":GET #3,C:? "} ± S Saves screen":IF C = 5 THEN 12
6 GET #3,C:IF C
19 THEN PUT #2,C:GOTO 6
8 C = PEEK(560) + 256*PEEK(561) + 4
9  POKE  850,A + 3-(A = 9):POKE  852,PEEK(C):POKE  853,PEEK(C + 1):POKE 856,192:POKE 857,3
10 C = USR(ADR("hhh*LVd"),16):IF W THEN 5 (* and d in reverse video!)
11 GET #3,C:IF PEEK(851) = 1 THEN 8
12 ? "#":END
```

LN 1 opens the keyboard and screen editor so we can create a text screen. It also gives us the ever popular green screen.

LN 2,3 find out if we want to read or write a screen. An "R" returns an 82 so we set A = 4 for READ in our open statement. We also set a flag W = 0 which we use later to indicate we are not writing successive screens. If the keyboard returns anything other than an 'R', then we set A = 8 for a write and set the write flag to W = 1. If the keyboard input is 'A' for APPEND, we set A = 9 for the OPEN statement. (Remember, this code is cryptic because I'm trying to squeeze a complete screen editor into less that 1K bytes.)

LN 4 OPENs FILE #1 for our burst IO. So the start of our IOCB for this file will be 834 + 16, or location 850.

LN 5 issues the prompt to remind the user of the control characters used to create a screen, edit a screen or save the screen.

LN 6 demonstrates the power of the Atari operating system. This single line allows us to create an entire text screen using all of the editing features of the OS screen editor: enter, backspace, delete, insert, move the cursor, etc.

LN 8 finds out where the pointer for the start of our screen data is so we can use the location in our busrt IO.

LN 9 sets the IOCB command for a read or write, depending on our input in LN 2,3. POKE 852,PEEK(C):POKE 853,PEEK(C + 1) set the start location of our screen data. POKE 856,192:POKE 857,3 tell the IOCB how many bytes to transfer (The BASIC manual tells us a Graphics 0 screen has 960 bytes. (3x256 + 192).)

LN 10 initiates the burst IO to either read or write a screen to disk. If we are writing screens, we go back to create the next screen.

LN 11 waits for a keyboard input if we are reading screens. Following any keyboard input, we go read the next screen if the CIO returned a 1 for a status code. This means A-OK. If it runs into an end-of-file, we get a different code and fall through to LN 12.

LN 12 clears the screen, and closes the file. If your disk has MENU on it, use RUN "D:MENU" instead.

How about saving any screen in any graphics mode, including modified display lists? OK. First, recall that the screen display data is saved right at the top of memory. Location 106 always points to the top of memory. (Sometimes we lie to the operating system and tell it that top of memory is lower than it really is. This lets us steal some memory from BASIC for our own machine language routines or character sets or player-missile data, etc.) The very top of memory holds the text window for our screen if we used one. Then the rest of the screen data is saved below the window. Finally, the display list is saved immediately below the screen data. So-o-o-o-o. Why not save the display list along with the screen data? This way we can handle modified display lists also. (Thank you, Sheldon Leemon, from the Michigan ACE.)

If we look at the pointer in 106, we have the start of our data for burst IO. If we subtract the start of the display list from the top of memory, we have how many bytes to transfer. Might as well save all of the color registers and the graphics mode also. We can do this by discrete PUTs before we start the burst IO.

When we retrieve our screen, the complete graphics mode will be set up the same as when we saved the screen. Better save the contents of 106 also. (If we're writing a program for another machine, we should tell our program the top of memory is as low as we can and still fit our program in. That way, our program might run on machines with less memory than ours.)

Here's one approach to a general purpose screen save/load routine: Note we are going to save the display list also. Since the display list points to where the screen data is, we must first POKE 106 with the lowest memory which still permits our program to run. This will make the saved display list compatible with smaller machines. If we get this general purpose, we will do well to save and restore the contents of 106 also. LN 32500 saves the screen; LN 32510 retrieves the screen.

```
1 ? "}D:SCRSAVE.LST":? :? "Save any mode screen to disk.":? "Load any mode screen from disk.":?
2  ?  "(Saves  all  colors  +  display  list.)":?  :?
3 ? "GOSUB 32500 to save screen.":? "GOSUB 32510 to load screen.":? :?
   "Send FILE$ = D:filename.DAT"
4 ? "MODE = Graphics mode of screen"
5 ? "Must POKE 106,128 to be compatible with 32k machines.":? :STOP
10 REM D:SCRSAVE.LST Save/retrieve any mode screen with display list & all color registers

32500 RAMTOP = PEEK(106)*256:DLIST = PEEK(560) + 256*PEEK(561):
BYTES = RAMTOP-DLIST:HI = INT(BYTES/256):LO = BYTES-HI*256
32502 CLOSE #5:OPEN #5,8,0,FILE$:PUT #5,MODE:PUT #5,PEEK(560):PUT #5,PEEK(561)
32503 FOR II = 704 TO 712:PUT #5,PEEK(II):NEXT II:IOCB = 834 + 5*16
32504   POKE   IOCB,11:POKE   IOCB + 2,PEEK(560):POKE
IOCB + 3,PEEK(561):POKE IOCB + 6,LO:POKE IOCB + 7,HI
32506 SCRSAVE = USR(ADR("hhh*LVd"),5*16):CLOSE #5:RETURN (reverse * and d)
32510 CLOSE #5:OPEN #5,4,0,FILE$:GET #5,MODE:GRAPHICS MODE:GET #5,A:POKE 560,A:GET #5,A:POKE 561,A
32511 FOR II = 704 TO 712:GET #5,COLR:POKE II,COLR:NEXT II
32512 IOCB = 834 + 5*16:POKE IOCB,7:POKE IOCB + 2,PEEK(560):POKE IOCB + 3,PEEK(561):POKE IOCB + 6,255:POKE IOCB + 7,255
32514 SCRLOAD = USR(ADR("hhh*LVd"),5*16):CLOSE #5:RETURN (reverse * and                                                        d)
```

The idea when retrieving a screen is to replicate the graphics status of the machine when you saved the screen.

Now you can generate, edit, save and retrieve PM data or character sets. Go modify those FONDEDIT or SUPERFONT programs to eliminate the DATA statements and save the complete character sets directly. Go ahead and save strings longer than 256 bytes. You know the address of the string from the ADR function and how many bytes it takes from the LEN functon so you're all set for BURSTIO! Machine language routines for page six will load in an instant. Once you get them entered the first time, simply burst them out to a disk file for future use. Speedy. Elegant!

Ron Miller's
FORTH AVENUE

GRAPHICS

Blinking Cursor in FORTH

The Atari operating system contains some countdown timers which are decremented 60 times a second during video vertical blank. When any timer hits zero, an interrupt is generated and an interupt service routine is executed. I am using one of these timers to periodically change the Atari cursor control register from normal video to inverse video and back. This is a FORTH version of a routine published in COMPUTE!, December, 1981. The routine was called "Blinking Characters" by Frank Jones.

The following FORTH code defines four FORTH words. BLINK is the interrupt service routine which blinks the cursor and resets the countdown timer. START.BLINK loads the timer the first time which starts the blinking.INIT.BLINK stores the PFA of BLINK (the ISR) into the ISR vector (228 hex). BC is just a nice short command to execute everything.

Once BC is executed, the Atari cursor will blink without any interference to the main program, which in this case if FORTH. Note that BLINK does not contain the normal "NEXT JMP," to return to FORTH. This is because BLINK is never executed by FORTH. It is executed by the Atari operating system during vertical blank.

For those of you who would like a detailed discussion of how this routine really works, come to the next North County Meeting and I will explain it.

```
( BLINK CURSOR ROUTINES )
HEX
CODE BLINK ( --- )
   02F3 LDA,   ( CUR CNTL REGS )
   2 # AND,    ( MASK BIT 2    )
   2 # EOR,    ( FLIP BIT 2    )
   02F3 STA,   ( CUR CNTL REG  )
   15 # LDA,   ( 1/2 SEC BLINK )
   021A STA,   ( TIMER REGSTR  )
   RTS,

( BLINK FLIPS CURSOR VIDEO )
( AND RELOADS COUNT DOWN   )
( TIMER WITH 21/60TH SECS  )

DECIMAL
( BLINK CURSOR SCR 2 )
HEX
CODE START.BLINK
   15 # LDA,   ( LOAD TIMER )
   021A STA,   ( 1ST TIME   )
   NEXT JMP,

: INIT.BLINK   ( INSTALL ISR )
    ' BLINK 228 ! ;

( BC WILL START BLINKING   )
( CURSOR..STACK NOT USED   )
: BC ( --- )
    INIT.BLINK START.BLINK ;

DECIMAL
```
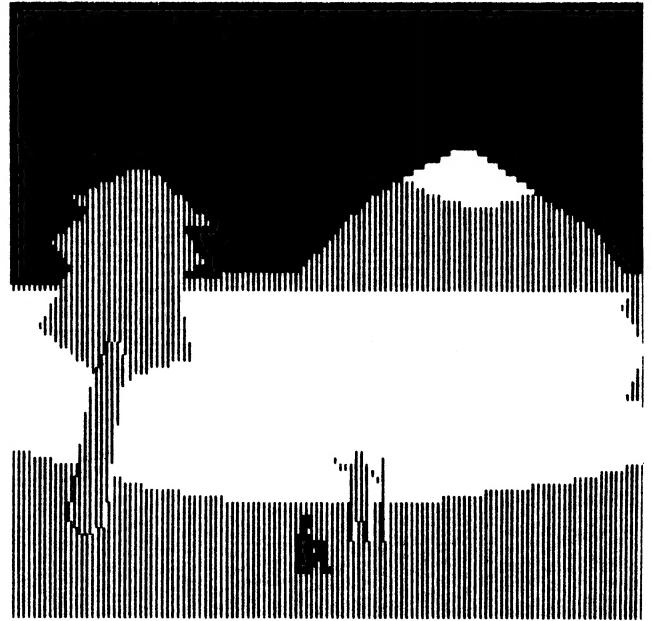
**A THREE DIMENSIONAL
"FILL" DUMP
BY FRITZ PORTER, S.D.A.C.E.**

**"Well, I used to be a paper shuffler,
but now I'm a number cruncher."**

BUY/SELL/TRADE

BUY/SELL/TRADE Advertisements are **FREE** to non-commercial members of the S.D.A.C.E. Ads must be 25 words or less. Specify the category desired. Only typewritten submission will be printed. (Give my eyes a break!) Mail to **B/S/T, 4037 Honeycutt Street, San Diego, CA 92109. No Phone Calls, Please!** Deadline is the 28th.

ETC.

You remember the Atari booster buttons? Hey, the fun we had with the E.T. dolls? I'll never forget the laughs I had with my Atari joy buzzer. And I never go anywhere without my Official Atari decoder ring. Now this:

ATARIAN HANDSHAKE

1. Clasp hands as in a regular handshake.

2. Lift the other person's thumb and grasp it with your free hand.

3. Rotate the thumb you're holding, making believe it's a joystick.

4. Enjoy.
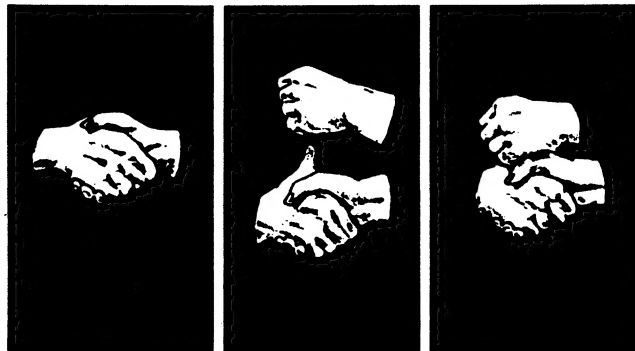
Thanks to the Atari Boosters League East - Winter Park, Florida, for this gem.

The I/O Connector

**5353 BALTIMORE DR., #39
LA MESA, CA 92041**

OUR NEXT MEETING,
21 February, 1983
6:30 p.m.

**North Park Recreation Center
4044 Idaho Street
Social Room**

•**Computer Expo Debriefing**
•**Atari Representative &**
•**Possible 1200XL Demo**

TO: